

# Live analysis and reconstruction of single-particle cryo-electron microscopy data with CryoFLARE

*Andreas D. Schenk<sup>1,\*</sup>, Simone Cavadini<sup>1</sup>, Nicolas H. Thomä<sup>1</sup>, Christel Genoud<sup>1</sup>*

<sup>1</sup>Friedrich Miescher Institute for Biomedical Research, Basel, Switzerland

\*Corresponding author: Andreas D. Schenk, [andreas.schenk@fmi.ch](mailto:andreas.schenk@fmi.ch)

## **Abstract**

Efficient, reproducible and accountable single-particle cryo-electron microscopy structure determination is tedious and often impeded by lack of a standardized procedure for data analysis and processing. To address this issue, we have developed the FMI Live Analysis and Reconstruction Engine (CryoFLARE). CryoFLARE is a modular open-source platform offering easy integration of new processing algorithms developed by the cryo-EM community. It provides a user-friendly interface that allows fast setup of standardized workflows, serving the need of pharmaceutical industry and academia alike who need to optimize throughput of their microscope. To consistently document how data is processed, CryoFLARE contains an integrated reporting facility to create reports.

Live analysis and processing parallel to data acquisition are used to monitor and optimize data quality. Problems at the level of the sample preparation (heterogeneity, ice thickness, sparse particles, areas selected for acquisition, etc.) or misalignments of the microscope optics can quickly be detected and rectified before data collection is continued. Interfacing with automated data

collection software for retrieval of acquisition metadata reduces user input needed for analysis, and with it minimizes potential sources of errors and workflow inconsistencies. Local and remote export support in Relion-compatible job and data format allows seamless integration into the refinement process. The support for non-linear workflows and fine-grained scheduling for mixed workflows with separate CPU and GPU based calculation steps ensures optimal use of processing hardware. CryoFLARE's flexibility allows it to be used for all types of image acquisitions, ranging from sample screening to high-resolution data collection, and offers a new alternative for setting up image processing workflows. It can be used without modifications of the hardware/software delivered by the microscope supplier. As it is running on a server in parallel to the hardware used for acquisition, it can easily be set up for remote display connections and fast control of the acquisition status.

## Introduction

The advent of direct electron detectors<sup>1-3</sup> in combination with data acquisition automation (e.g. with EPU, SerialEM<sup>4</sup>, Leginon<sup>5</sup>) and new algorithmic advances in image processing have contributed to the establishment of single-particle cryo-EM as a successful and attractive tool for structure determination of proteins and protein complexes<sup>6,7</sup>. To obtain high resolution cryo-EM structures, data in the order of terabytes are routinely collected, analyzed and processed. The large amount of data produced is a growing challenge as it must be shared, stored, transferred, viewed and archived. To overcome this challenge automated tools for quality assessment, data processing, data handling and reporting are required. In parallel, it is necessary to keep hardware and software updated in order to follow the fast developments in the single-particle cryo-EM field. Therefore, a data processing workflow needs to be fully open and easy to modify in order to integrate newly developed algorithms at the pace they are offered to the community.

In this context we introduce CryoFLARE to perform live analysis and reconstruction of data in parallel to data acquisition. It is an open-source software written in C++/Qt5. It shares similarities with processing frameworks as e.g. Focus<sup>8</sup> or Warp<sup>9</sup> but focuses more on providing a standardized

and yet flexible graphical workflow with strong integration of data acquisition and reporting suitable to document the data integrity prior to publication. CryoFLARE does not require programming knowledge for setup and use. It provides an intuitive user interface displaying multiple parameters during acquisition such as ice thickness, devitrification level and number of particles picked. Acquisition parameters as e.g. camera type, nominal defocus, phase plate information and exposure time are automatically imported from the acquisition software without user input, ensuring coherence and accuracy of acquisition metadata.

CryoFLARE is agnostic to the actual algorithms used for processing. Instead it provides an interface to communicate with existing processing scripts or programs by transferring parameters through standard input and output channels.

CryoFLARE allows flexible integration of newly available algorithms instead of depending on a fixed pipeline or implementation. Its real-time visualization capabilities allow live monitoring of data acquisition and can serve as a tool for ensuring microscope performance and data quality.

CryoFLARE provides a fast and rigid testing ground for evaluation and comparison of new processing strategies or algorithms. It supports optimization and acquisition with Volta phase plates (VPP) by monitoring and controlling the positions used on the VPP and tracking its performance

Users can assess the quality of their data and adapt the image acquisitions conditions quickly at the beginning of their imaging sessions. Micrographs to be kept and re-processed can be selected interactively. CryoFLARE can also transfer the data to the image analysis software of choice easily. For the experts in image analysis responsible for the data acquired in a laboratory, the clear and documented structure of CryoFLARE allows a fast adaptation of the workflow to special processing needs, e.g. switching to different CTF determination software for tilted data collection, or automatic phase shift fitting if a phase plate is inserted during acquisition. The graphical user interface makes users aware of the micrograph quality and data acquisition optimization as the feedback is instantaneous and displayed graphically. Its intuitive design allows going through micrographs and

adapting particle picking strategies while data is acquired. The detailed aspects are illustrated and described below.

## User interface

The graphical user interface (Figure 1) is organized into two separate pages. The first page is dedicated to visualizing micrograph related results, while the second page is used to aggregate information on the grid square level.

The micrograph page consists of multiple areas: The table in the top left (Figure 1B) lists each acquired micrograph and displays the corresponding metadata from the acquisition software (e.g. pixel size, dose, etc.) live as well as metadata created by processing tasks (e.g. defocus, drift, etc.). The design of the table was inspired by the IPLT diffraction processing pipeline<sup>10-12</sup>. The table also serves as a tool for selecting individual micrographs or micrograph groups for export. The panel on the right-hand side (Figure 1C) shows details about the micrograph currently selected in the micrograph table. The results are grouped by processing task (e.g. motion correction, particle picking, etc.) and consist of metadata or images and plots associated to the task. Each task is displayed in a panel and all panels are organized as tabs that can be selected. In addition, the panel allows modifying the input parameters for the displayed task. For example, in the particle picking panel, the picking algorithm and box size can be modified at any time during processing. The bottom panels (Figure 1D-F) provide live graphs calculated over the whole data set. From left to right the first graph shows a linear plot, following the progress of the data acquisition with the X-axis representing the micrograph number. The parameter to be plotted in Y can be chosen by selecting a parameter column in the micrograph table (for example astigmatism). The second graph shows a histogram distribution of the same selected parameter. The third panel shows a heat plot and allows mapping the selected parameters onto individual VPPs and VPP positions, if the dataset was acquired with a VPP present. The graphs can be freely zoomed and are updated in real time with

each processed micrograph. In addition, graphs can be used for interactive selection of micrographs by the user (see Data selection below). Selected micrographs are plotted in blue, while deselected micrographs are plotted in red. In addition to the standard plots, two freely selectable parameters can also be plotted against each other in a scatter plot to evaluate their correlation. For example, we found it useful to plot ice thickness versus the maximum resolution of the CTF fit, as thinner ice often correlates with better data quality(Figure 4C).

The grid square page shows an overview of the recorded grid squares. The overview micrograph can be overlaid with a heat plot, mapping any of the results onto their relative foil hole position within the grid square.

The processing of each micrograph is split into separate tasks for each step of the processing (drift correction, CTF determination, etc.). The processing tasks to be performed on each micrograph can be defined in the settings dialog box. The tasks are organized into a tree instead of a linear workflow to allow parallel execution of independent processing tasks. For each task a set of input and output parameters can be defined to facilitate data exchange with the processing script or program.

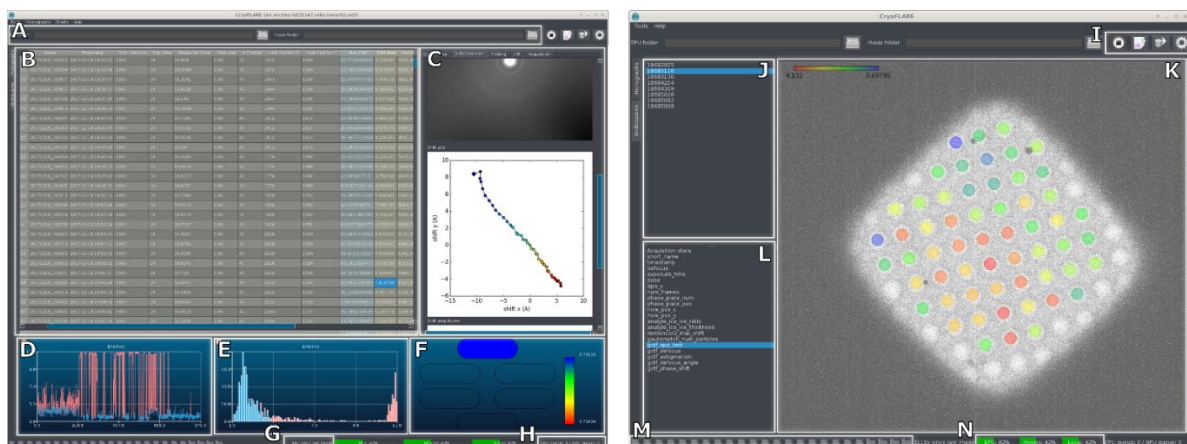


Figure 1: Graphical user interface: (A) Project and movie folder selection. (B) List of micrographs displaying results from acquisition and processing. (C) Detail panel showing results from currently active micrograph as well as input form for user defined parameters. (D) Linear plot. (E) Histogram. (F) Phase plate heat plot. (G) Counter indicating time since last acquisition. (H) CPU and GPU queue (I) Buttons for export, reporting, start/stop and settings. (J) Grid square selection. (K) Grid square

display with heat plot overlay. (L) Parameter selection for heat plot. (M) CPU and GPU activity display. (N) Disk usage display.

## Features

### *Import*

CryoFLARE supports three separate import options to handle differences in folder organization by different data acquisition packages. Data can either be organized in 1) an EPU folder structure, with full support for multiple Images-Discs and grid squares, 2) a flat folder structure providing micrographs and EPU metadata xml files in a single folder, allowing reimport of already processed EPU data, or 3) a flat folder structure with optional metadata in json format, allowing processing of data from 3<sup>rd</sup> party data acquisition packages as e.g. SerialEM<sup>4</sup> or Legion<sup>13</sup>. Extracting and using metadata directly from the data acquisition software allows users to concentrate on data analysis as it avoids the burden of manually entering already determined parameters again into the analysis software. This automation reduces the source of possible errors by misentering data. For EPU data, CryoFLARE reads the metadata associated to grid squares foil holes as well to allow data analysis on the grid square level.

CryoFLARE supports file creation and modification detection on remote shares (e.g. CIFS or NFS), hence enabling direct import from a remote location as for example a microscope or camera PC. Support for remote data locations allows to separate the microscope network segment (in general accessible to external parties for remote support) from the internal network environment which might contain confidential data (Figure 6B). Fencing off the data collection hardware reduces the attack surface on the internal network and lowers the risk of exposure of internal data.

## Data selection

To provide an optimal data set for further processing it is invaluable to have a selection regime in place to remove bad micrographs during live processing. CryoFLARE supports the user by providing three easy to use selection modes. 1) The list of micrographs can be sorted based on any of the results and then micrographs can either be selected individually or all micrographs above/below a certain limit can be selected or deselected. 2) Subsets of micrographs can be interactively selected (Figure 2A) in a live plot. 3) Micrographs can automatically be selected or deselected based on an upper and lower limit for a parameter within a processing script (Figure 2B, Supplementary Figure 2A). The ability to select a subset of good data for subsequent export leads to faster data transfer, reduced storage requirements and facilitates classification and refinement, therefore supporting a sustainable high quality and high throughput processing workflow.

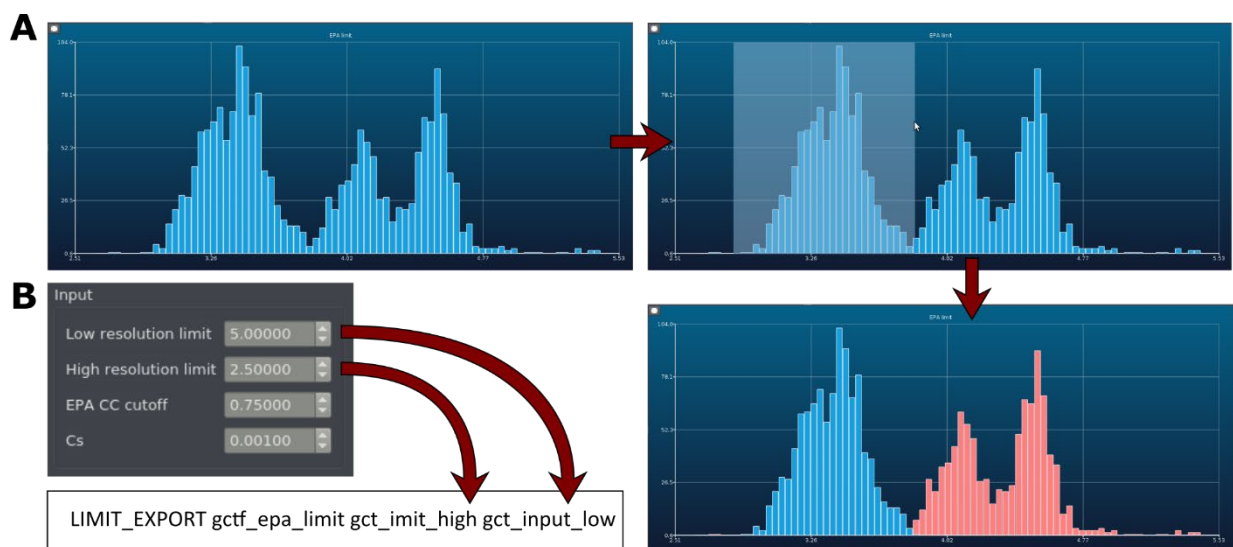


Figure 2: Data selection: (A) Data can be interactively selected with the mouse on the real time graphs. (B) Data can be selected based on user defined limits, which are then used for selection within the processing script. The shown example selects all images that show a GCTF EPA limit (stored in the bash variable `gctf_epa_limit`) between 2.5 Å and 5.0 Å for export.



## Reporting

For a single-particle EM structure determination pipeline, especially if it is run in a facility setup, it is becoming increasingly important to provide adequate documentation on data acquisitions and processing steps that are performed. The documentation provides accountability and project continuity by simplifying handover of data between multiple people involved in a project. CryoFLARE supports this effort by providing a reporting facility that can be tailored to individual use cases. Report templates can be assembled in a graphical user interface (Figure 3A), where parameters or images from data acquisition or processing can be selected interactively. At the end of the data acquisition a report in pdf format (Figure 3B) can be created using the template assembled beforehand.

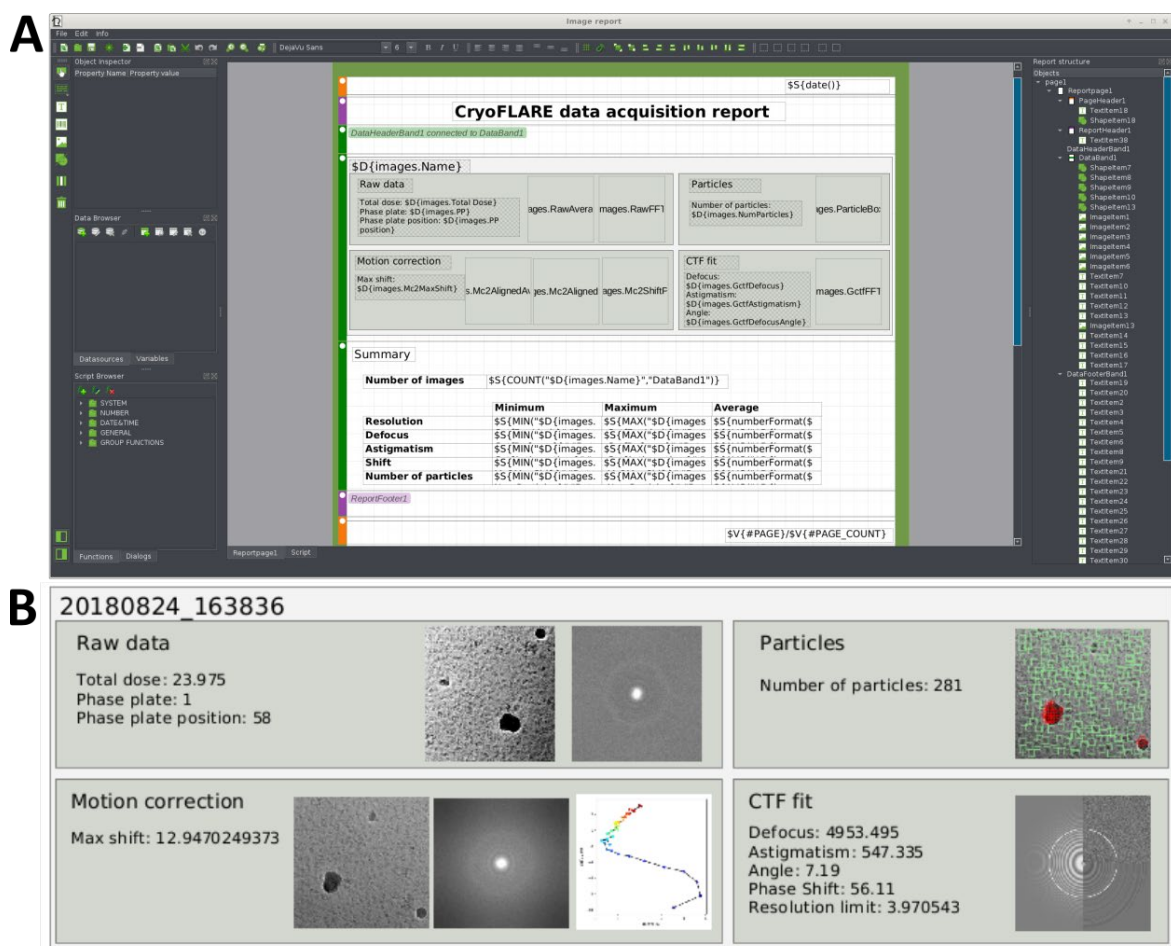


Figure 3: (A) Report builder. (B) Sample report output for single micrograph.



In addition, the reported values can also be saved as a list of comma-separated values (CSV) or in json format. The saved files allow creation of graphs for publication in external software, as illustrated in the example plots in Figure 4B and C.

## *Processing*

For each acquired micrograph a set of processing tasks is performed. The metadata generated during acquisition and processing is carried over to subsequent tasks, making it easy to set up processing pipelines with minimal user interaction. The metadata together with the processing settings are stored permanently within the processing folder to ensure persistence of the data across multiple program executions.

The CryoFLARE program itself does not include any processing algorithms or user interface elements fixed to a certain metadata parameter. Instead it provides an API (see Scripting interface) to dynamically execute user defined processing tasks. This design was chosen to allow users the flexibility to define their own workflows and to enable them to include newly developed processing algorithms, instead of forcing users to use one specific implementation. The workflow is saved and loaded together with the rest of the settings. CryoFLARE saves the settings within the processing directory and opening CryoFLARE again in the same directory at a later stage will load them again to restore the workflow. In addition, settings can be saved as account wide default, which are then used whenever CryoFLARE is started in a processing directory not containing any settings.

Using this generic interface, we set up a processing workflow which is distributed together with CryoFLARE, containing the following advanced features:

**Multi-step CTF determination** : The CTF determination script uses Gctf<sup>14</sup> to determine the CTF parameters in an in-house developed two-step process. In a first step a fit is performed with a defocus step size of 500 Å. The search range is dependent on the nominal defocus, as provided by

the data acquisition software. A lower limit of half the nominal defocus and an upper limit of 2.5 times the nominal defocus is used. In a second step, the CTF parameters are refined using a defocus step size of 100 Å within a range of 0.9x to 1.1x of the average defocus determined in step one. For equiphase averaging resolution estimation of the final fit, a user defined cross correlation cutoff can be set. A default cutoff value of 0.75 is used. The resulting CTF parameters are stored into a star file for further use in 2D classification in Relion<sup>15</sup> or Cryosparc<sup>16</sup>. The CTF parameters are also used to deconvolute the micrographs (Figure 5D) using a script implemented in IPLT<sup>10,11</sup> following the algorithm described in Tegunov & Cramer<sup>9</sup>.

**Particle picking:** Particles are picked on each micrograph using Gautomatch (<https://www.mrc-lmb.cam.ac.uk/kzhang/>). The Gautomatch output is parsed and particle positions are indicated on the micrograph in green while rejected positions are indicated in red (Figure 5C).

**Ice thickness determination:** The ice thickness of each acquired micrograph is automatically estimated using the aperture limited scattering methodology laid out in Rice et al.<sup>17</sup>. The python script performing the calculation uses EMAN2<sup>18</sup> to extract the relevant micrograph data. The sole input that must be provided by the user is a reference micrograph without a sample recorded with identical illumination as used for imaging.

**Devitrified ice detection:** Devitrification is detected using a homemade script by calculating the ratio between the signal intensity in a micrograph at 3.89 Å and 5.0 Å. Deviation of the ratio from 1.0 can signal possible devitrification of the sample. The python script uses EMAN2<sup>18</sup> to calculate the rotational average of the Fourier transform of the micrograph. From the rotational average an ice band at  $3.89 \pm 0.2$  Å and a reference band at  $5.0 \pm 0.2$  Å are extracted. The maximum value in the ice band ( $V_{ice}$ ) and the reference band ( $V_{ref}$ ) are determined and the ratio  $V_{ice}/V_{ref}$  is calculated. A ratio significantly over 1.0 reproducibly indicates micrographs with devitrified ice.

**Amplitude graphs and 2D plots for motion correction:** The script provided with CryoFLARE for motion correction uses MotionCor2<sup>19</sup>. To visualize the correction 2D scatter plots depicting the

frame to frame shift vectors (Figure 4A) and amplitude plots depicting the absolute amount of shift between frames are created using matplotlib<sup>20</sup>.

**Tracking of acquisition position:** CryoFLARE reads the EPU metadata files for each acquired micrograph to extract the actual position of the acquisition in reference to the grid square. The position is drawn as a green circle on the grid square overview image.

**Aberration present image shifting support:** CryoFLARE creates a 2D vector diagram with matplotlib, indicating the applied image shift for each acquisition. This information can be used to identify individual acquisitions in EPU template setups used for aberration present image shifting<sup>21</sup> (APIS). In addition, CryoFLARE tracks and writes template position ID into the star file containing the CTF information for each micrograph, therefore allowing grouping by template position for beam tilt refinement (e.g. in Relion).

**Data aggregation:** To monitor performance of the instrument over time, each exposure is logged in csv format in a file. For each exposure the sample position, phase plate position (if a phase plate was used), maximum drift, ice thickness and quality, CTF parameters including phase shift (if applicable) and number of detected particles are stored. The csv data can be used for plot generation as for example shown in Figure 4D.

**2D classification:** Automatic 2D classification can be triggered once a user-defined number of particles are picked. The 2D classification will then be carried out on particles from selected micrographs (see Data selection) by filtering out deselected micrographs from all star files involved. The particle coordinates determined during particle picking are used to extract particle boxes with `relion_preprocess`. The extracted particles are then classified in 2D using `relion_refine`. The results are displayed as a montage created by Sparx/EMAN2.

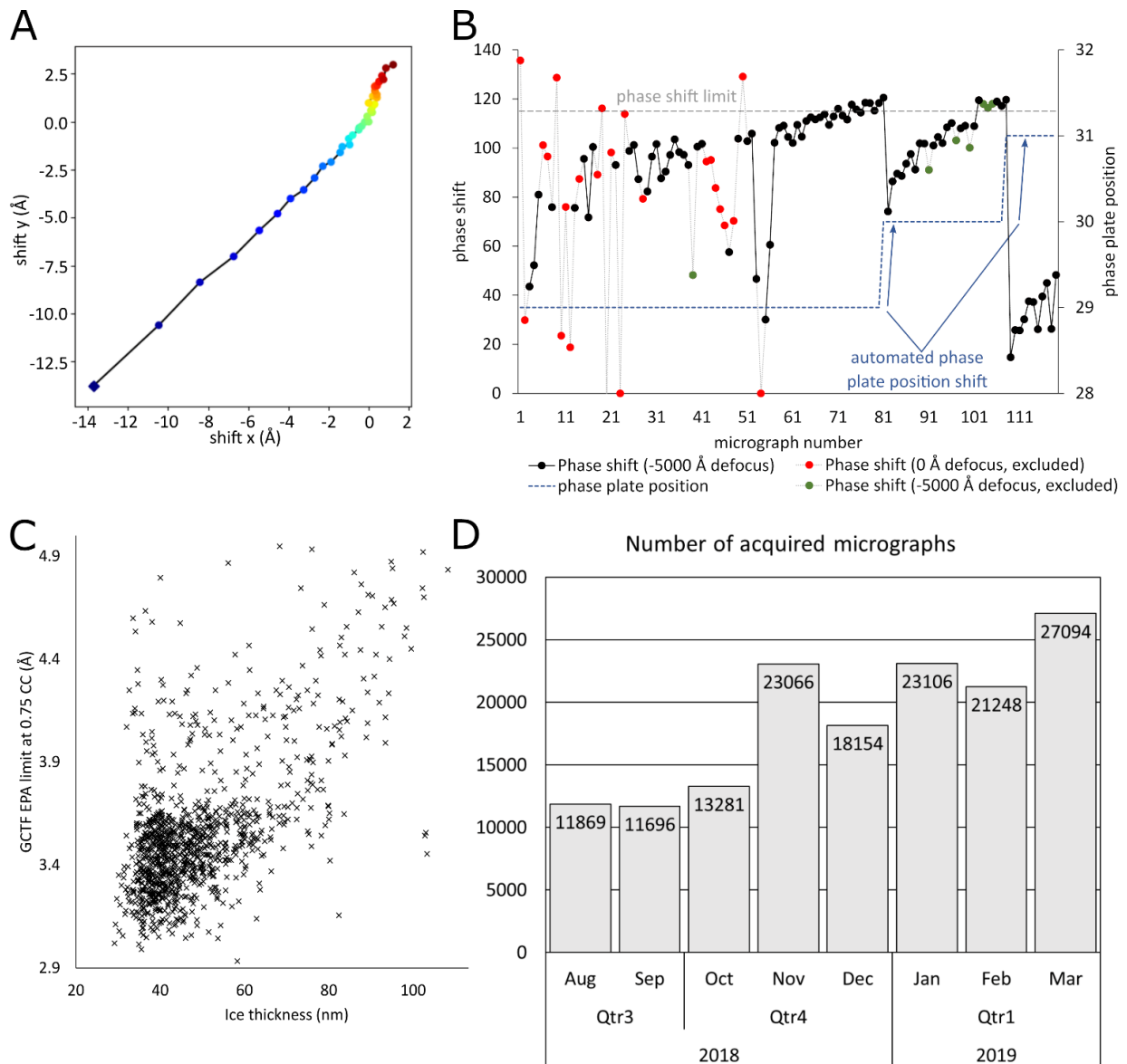


Figure 4: (A) XY plot of frame shifts during motion correction with a blue diamond marking the first frame. (B) Phase shift of individual micrographs. Micrographs showing problems during CTF fitting are automatically excluded. Most micrographs were excluded due to low contrast because of in focus acquisition (red), whereas few micrographs showed other fitting problems (green). The phase shift values were used to automatically control phase plate movement. The phase plate position was shifted at a median phase shift above 115° within 10 micrographs. (C) Scatter plot correlating ice thickness to GCTF EPA limit. (D) Number of acquired micrographs over a time span of 8 months.

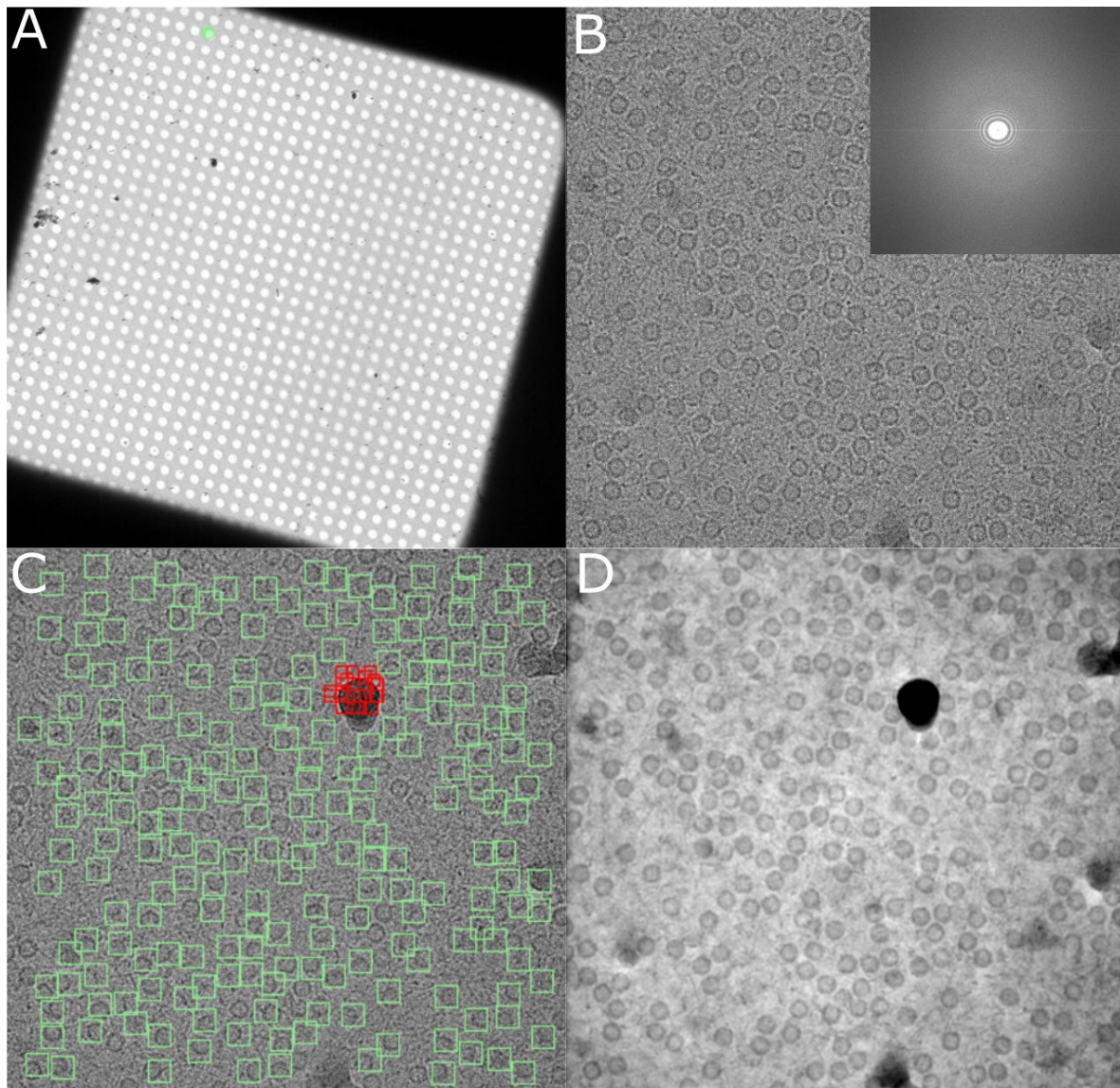


Figure 5: Sample workflow: (A) Foil hole position of recorded micrograph indicated in green on grid square. (B) Motion corrected micrograph with FFT in inset. (C) Green boxes indicating picked particles. Red boxes indicating excluded areas. (D) Deconvoluted micrograph.

### *Feedback control*

CryoFLARE provides a Windows service to enable access to Advanced TEM Scripting from a remote computer, where CryoFLARE is running. This is used in our setup to automate changing of the phase plate position. A script checks the CTF fits of a user defined number (10 in this example) of the most recent micrographs, and once the median phase shift exceeds a user defined threshold ( $115^\circ$  in this example), the phase plate is shifted to the next position (Figure 4B). Automation of phase plate

movement ensures that micrographs are recorded in an optimal phase shift range to maximize micrograph contrast.

## *Export*

The export of the data is split into two separate elements: 1) the file and destination selection and 2) the data filtering and actual transfer. The file destination selection provides the user with a dialog, where he can select which files to export for the chosen micrographs. The files are split into three separate lists for raw data, processing data, and shared files (e.g. star files listing all micrographs). The user can also browse for a local or remote destination folder as well as a separate destination folder for the raw data (e.g. for separate backup). Any folder reachable by ssh can be browsed and selected as destination.

Once the data transfer is started, the exporter will generate a list of micrographs with associated processing data to be transferred. Only data from selected micrographs that are fully processed and selected by the user, or the processing script, will be transferred. As a second step the exporter parses all star files containing data from multiple micrographs and filters out micrographs that are not exported. The filtering step is an important part of the workflow to ensure only useful data is exported and thereby reducing transfer times and storage requirements.

The selected data is exported to the destination using several threads in parallel to optimize data transfer speed.

## **Sample workflow**

Here we present a sample workflow in CryoFLARE as it is currently used in our facility. An overview of the workflow can be seen in Figure 6A. The first task is to transfer the averaged micrograph and metadata from the microscope PC and the raw stack and gain reference from the Gatan PC (for a K2 camera) or Offload PC (for a Falcon III camera) to the processing server. Once the transfer is



complete, as next steps motion correction, ice analysis and visualization of the acquisition position followed by image shift visualization are performed in parallel. Visualization of inter-frame motion is performed in a script separate from the actual motion correction to avoid blocking GPU resources longer than necessary. The visualization is run in parallel with the next GPU task, which is the CTF determination. Upon completion, a feedback task is started allowing automatic skipping to the next phase plate position once a phase shift threshold has been reached. In parallel, processing is continued with GPU accelerated particle picking. Once particle picking is finished the visualization of particle positions is performed and at the same time 2D classification is started once enough particles were picked.

## Implementation

CryoFLARE is organized into three different layers (Figure 6B). The interaction layer handles the interaction with the user, the acquisition software, backup locations and further downstream processing like classification and refinement. The management layer is responsible for scheduling the processing and optimizing the use of the computing infrastructure. The processing layer oversees running the processing tasks and interfacing with the external processing scripts and programs.

### *Scheduler*

At the core of the CryoFLARE implementation sits the scheduler. The scheduler contains two stacks to separate scheduling of GPU and CPU tasks. The tasks are processed in last-in-first-out (LIFO) order to give highest priority to the most recent data. The number of processing slots assigned for each stack can be configured separately, as on most systems the number of CPU cores does not match the number of GPUs. This setup allows to fully use the CPU resources without forcing multiple tasks to share the same GPU, given that this could lead to instabilities in the processing (e.g. with MotionCor2 version 1.0.1 and above). Breaking up the micrograph processing into separate



tasks allows to define non-linear workflows and with it parallelization within a micrograph processing workflow (Figure 6A). In addition, the task-based scheduler allows allocation of GPU resources to be limited to tasks utilizing it, instead of allocating the resources for the full workflow. Tasks are spawned as separate processes using the QProcess interface.

The scheduler checks the completion of each tasks and tracks that information by setting flags in the metadata object (see below) associated with a micrograph.

The organization of the scheduling into self-contained tasks together with the fact that CryoFLARE also reads in metadata for grid squares would also allow to run tasks on the grid square level in the future to e.g. predict good sample areas before data acquisition and transfer back that information into the data acquisition software.

## *Metadata*

Metadata is internally stored as Json object. The metadata associated with each micrograph consists of data acquisition parameters, result parameters generated during data processing within CryoFLARE, file paths for files generated during processing as well as completion states of the processing tasks. Upon each change of metadata, the update is written out to a binary encoded Json file in order to ensure data persistence. Upon restart of CryoFLARE in a pre-existing processing directory the binary encoded metadata is read in to restore the processing state.

## *Scripting interface*

A scripting interface is provided within CryoFLARE to transfer metadata between user interface and processing scripts or programs. Upon start of a processing task by the scheduler all existing metadata items for the micrograph to be processed are transferred to the processing script/program as key=value pairs on the standard input. For python and bash script an adapter is provided that automatically translates the input into script variables named after the metadata keys

(Supplementary Figure 1A). The user interface settings allow definition of additional metadata items to serve as user defined input for processing tasks.

Output from the processing is transferred back to CryoFLARE in a similar way on the standard output. In addition to output parameters, files can be defined as well. Three types of files are distinguished: output files, raw data files, and files shared between all micrographs (e.g. micrographs star files). The distinction enables more selective data export (see Export). For bash and python scripts convenience functions are provided to easily define output parameters (Supplementary Figure 2C) and files (Supplementary Figure 1B,C). It is important to note that any output metadata is automatically made available as input to subsequently running child tasks.

To facilitate interaction with Relion for subsequent processing, a set of scripting functions is provided to create Relion job folders and pipeline files as well as creating and adding data to star files (Supplementary Figure 2B). The scripting functions take care of file locking in order to make them safe to use during parallel processing of multiple micrographs.

### *Feedback API*

To transfer back information and control the microscope CryoFLARE provides a bridge to Advanced TEM scripting. The bridge consists of a Windows service implemented in Python using the pywin32 interface. It takes commands from a configurable control file, which are then translated and passed on to the Advanced TEM scripting COM interface, which is accessed through the comtypes module. Use of a file-based control connection allows to easily pass firewalls. CryoFLARE processing tasks can write commands to the control file (located on a mounted remote share) to control microscope function and therefore feedback processing information to improve data collection.

### *Importer*

The data import facility consists of two separate elements: a file and folder listener to watch for newly acquired data and a metadata reader to retrieve metadata information from the data

acquisition software. Both elements are encapsulated in an importer class to allow easy adaptation to different folder and metadata structures.

The file and folder listener is a drop-in replacement for QFileSystemWatcher, reimplemented from scratch using a polling scheme to support listening on remote shares that do not provide full file change notification support. The listener is started in a separate thread to avoid unnecessarily blocking the user interface and supports watching multiple folder within a directory structure in order to fully observe nested directories as created by EPU.

Two metadata readers are implemented, one for XML and one for optional Json metadata. The XML reader supports metadata as written by EPU and the optional Json metadata reader can be used to read in data from other automated acquisition software or from manual acquisitions.

### *Exporter*

The remote browsing and transfer are implemented using the Qssh library. Only data from selected micrographs that have been fully processed (determined by the flags set by the scheduler) will be transferred. Multiple processes, which open multiple ssh channels for remote transfer, are used to transfer the data in order to use the full bandwidth of the network connection to the remote storage.

### *Reporting*

The report creation in CryoFLARE is based on the Lime report generator framework (<http://limereport.ru>). The reporting engine was extended to allow reading image files from disk and integrate them into a report. The model containing the micrograph metadata is used as data source for the report generator, allowing direct selection of any metadata parameter configured in settings for reporting.

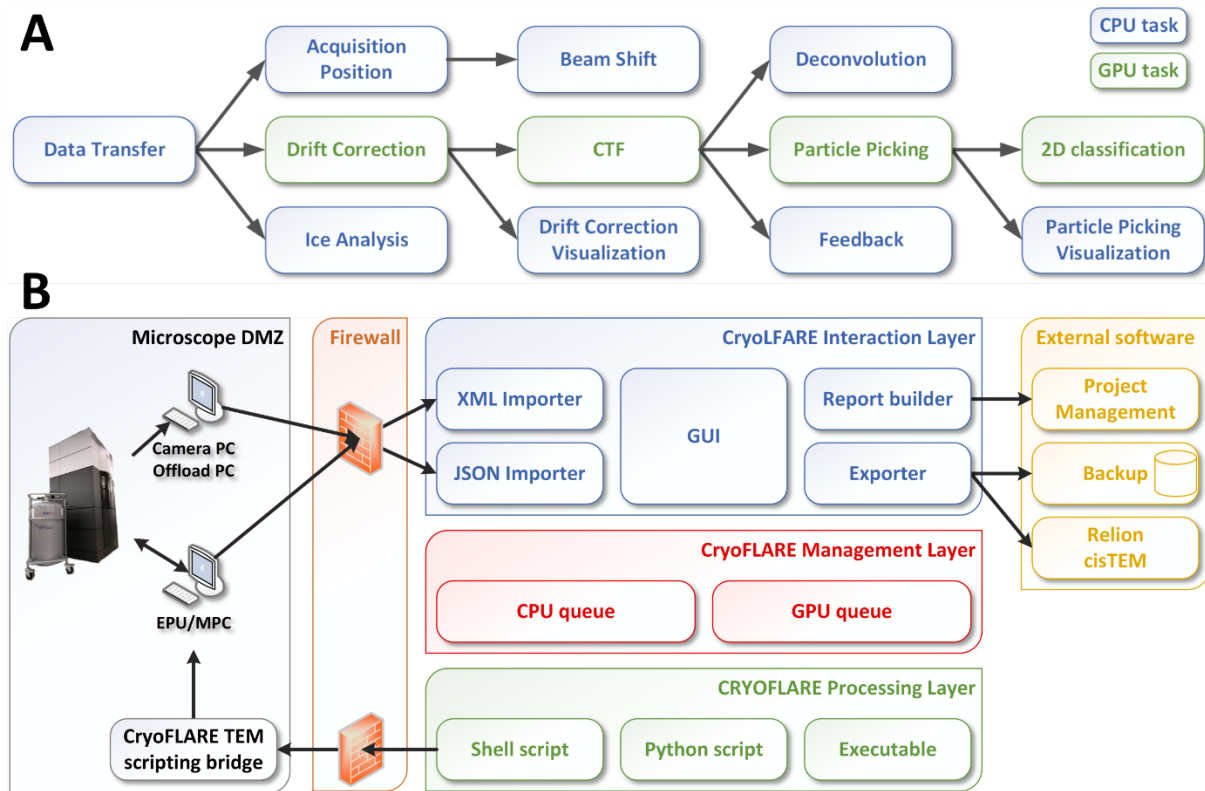


Figure 6: Implementation: (A) process tree. (B) Software layers.

## Discussion and Outlook

With the development of CryoFLARE we provide a tool for live assessment and optimization of data collection, which can help electron microscopy laboratories and facilities to increase the throughput of single-particle cryo-EM projects.

CryoFLARE is routinely used in our facility for data collection and already contributed to multiple cryo-EM structures (e.g. BK-VLP\_scFv<sup>22</sup>, nucleosome/DDB2<sup>23</sup>, BRISC<sup>24</sup>) at FMI and Novartis. In addition, it has been used by ThermoFisher for demonstrating data acquisition capabilities of their instruments.

The software allows for prompter and more in-depth data analysis of acquired data, supporting the microscope operator in improving the acquisition strategy and reducing the amount of low-quality data recorded.

The data aggregation and reporting facilities support the users in the documentation of their EM experiments and help to improve the accountability for EM facilities.

The feedback mechanisms included with CryoFLARE are a first step towards more data driven decisions, where the data acquisition strategy is optimized continuously based on the results of the data analysis. The flexibility and extendibility of CryoFLARE together with its tight integration to data acquisition make it an easy-to-use test system for new acquisition and processing strategies, which will help to drive the evolution of the single-particle cryo-EM field.

## **Software availability**

CryoFLARE binaries for Linux and documentation can be found at [www.cryoflare.org](http://www.cryoflare.org). The source code of CryoFLARE is licensed under the GNU General Public License (GPL) and is available from GitHub: [www.github.com/fmi-basel/faim-cryoflare.git](https://www.github.com/fmi-basel/faim-cryoflare.git).

## **Author contribution**

AS conceived, developed and tested the software. SC helped test the software. AS, CG and NT wrote the manuscript.

## **Acknowledgements**

We would like to thank Alexandra Graff-Meyer, Christian Wiesmann, Honnappa Srinivas, Céline Be and Maryam Khoshouei from the FMI/Novartis electron microscopy competence center for fruitful discussions. We are also very grateful for Adriano Marra for help with computing and Bart van Knippenberg, Mazdak Radjainia and Fanis Grollios for their collaboration in interfacing CryoFLARE with EPU.

## References

- (1) Kuijper, M.; van Hoften, G.; Janssen, B.; Geurink, R.; De Carlo, S.; Vos, M.; van Duinen, G.; van Haeringen, B.; Storms, M. FEI's Direct Electron Detector Developments: Embarking on a Revolution in Cryo-TEM. *Journal of Structural Biology* **2015**, *192* (2), 179–187. <https://doi.org/10.1016/j.jsb.2015.09.014>.
- (2) Li, X.; Zheng, S. Q.; Egami, K.; Agard, D. A.; Cheng, Y. Influence of Electron Dose Rate on Electron Counting Images Recorded with the K2 Camera. *J. Struct. Biol.* **2013**, *184* (2), 251–260. <https://doi.org/10.1016/j.jsb.2013.08.005>.
- (3) Milazzo, A.-C.; Cheng, A.; Moeller, A.; Lyumkis, D.; Jacovetty, E.; Polukas, J.; Ellisman, M. H.; Xuong, N.-H.; Carragher, B.; Potter, C. S. Initial Evaluation of a Direct Detection Device Detector for Single Particle Cryo-Electron Microscopy. *Journal of Structural Biology* **2011**, *176* (3), 404–408. <https://doi.org/10.1016/j.jsb.2011.09.002>.
- (4) Mastronarde, D. N. Automated Electron Microscope Tomography Using Robust Prediction of Specimen Movements. *J. Struct. Biol.* **2005**, *152* (1), 36–51. <https://doi.org/10.1016/j.jsb.2005.07.007>.
- (5) Carragher, B.; Kisseberth, N.; Kriegman, D.; Milligan, R. A.; Potter, C. S.; Pulokas, J.; Reilein, A. Leginon: An Automated System for Acquisition of Images from Vitreous Ice Specimens. *J. Struct. Biol.* **2000**, *132* (1), 33–45. <https://doi.org/10.1006/jsbi.2000.4314>.
- (6) Frank, J. Nature Protocols – 10 Year Commentary - Advances in the Field of Cryo-Electron Microscopy over the Last Decade. *Nat Protoc* **2017**, *12* (2), 209–212. <https://doi.org/10.1038/nprot.2017.004>.
- (7) Nwanochie, E.; Uversky, V. N. Structure Determination by Single-Particle Cryo-Electron Microscopy: Only the Sky (and Intrinsic Disorder) Is the Limit. *Int J Mol Sci* **2019**, *20* (17). <https://doi.org/10.3390/ijms20174186>.
- (8) Biyani, N.; Righetto, R. D.; McLeod, R.; Caujolle-Bert, D.; Castano-Diez, D.; Goldie, K. N.; Stahlberg, H. Focus: The Interface between Data Collection and Data Processing in Cryo-EM. *Journal of Structural Biology* **2017**, *198* (2), 124–133. <https://doi.org/10.1016/j.jsb.2017.03.007>.
- (9) Tegunov, D.; Cramer, P. Real-Time Cryo-Electron Microscopy Data Preprocessing with Warp. *Nat. Methods* **2019**, *16* (11), 1146–1152. <https://doi.org/10.1038/s41592-019-0580-y>.
- (10) Philippsen, A.; Schenk, A. D.; Stahlberg, H.; Engel, A. Iplt--Image Processing Library and Toolkit for the Electron Microscopy Community. *J. Struct. Biol.* **2003**, *144* (1–2), 4–12.
- (11) Philippsen, A.; Schenk, A. D.; Signorell, G. A.; Mariani, V.; Bernèche, S.; Engel, A. Collaborative EM Image Processing with the IPLT Image Processing Library and Toolbox. *Journal of Structural Biology* **2007**, *157* (1), 28–37.
- (12) Schenk, A. D.; Philippsen, A.; Engel, A.; Walz, T. A Pipeline for Comprehensive and Automated Processing of Electron Diffraction Data in IPLT. *Journal of Structural Biology* **2013**, *182* (2), 173–185. <https://doi.org/10.1016/j.jsb.2013.02.017>.
- (13) Suloway, C.; Pulokas, J.; Fellmann, D.; Cheng, A.; Guerra, F.; Quispe, J.; Stagg, S.; Potter, C. S.; Carragher, B. Automated Molecular Microscopy: The New Leginon System. *J. Struct. Biol.* **2005**, *151* (1), 41–60. <https://doi.org/10.1016/j.jsb.2005.03.010>.
- (14) Zhang, K. Gctf: Real-Time CTF Determination and Correction. *J. Struct. Biol.* **2016**, *193* (1), 1–12. <https://doi.org/10.1016/j.jsb.2015.11.003>.
- (15) Scheres, S. H. W. RELION: Implementation of a Bayesian Approach to Cryo-EM Structure Determination. *Journal of Structural Biology* **2012**, *180* (3), 519–530. <https://doi.org/10.1016/j.jsb.2012.09.006>.
- (16) Punjani, A.; Rubinstein, J. L.; Fleet, D. J.; Brubaker, M. A. CryoSPARC: Algorithms for Rapid Unsupervised Cryo-EM Structure Determination. *Nat. Methods* **2017**, *14* (3), 290–296. <https://doi.org/10.1038/nmeth.4169>.

- (17) Rice, W. J.; Cheng, A.; Noble, A. J.; Eng, E. T.; Kim, L. Y.; Carragher, B.; Potter, C. S. Routine Determination of Ice Thickness for Cryo-EM Grids. *Journal of Structural Biology* **2018**, *204* (1), 38–44. <https://doi.org/10.1016/j.jsb.2018.06.007>.
- (18) Tang, G.; Peng, L.; Baldwin, P. R.; Mann, D. S.; Jiang, W.; Rees, I.; Ludtke, S. J. EMAN2: An Extensible Image Processing Suite for Electron Microscopy. *J. Struct. Biol.* **2007**, *157* (1), 38–46. <https://doi.org/10.1016/j.jsb.2006.05.009>.
- (19) Zheng, S. Q.; Palovcak, E.; Armache, J.-P.; Verba, K. A.; Cheng, Y.; Agard, D. A. MotionCor2: Anisotropic Correction of Beam-Induced Motion for Improved Cryo-Electron Microscopy. *Nat. Methods* **2017**, *14* (4), 331–332. <https://doi.org/10.1038/nmeth.4193>.
- (20) Hunter, J. D. Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering* **2007**, *9* (3), 90–95. <https://doi.org/10.1109/MCSE.2007.55>.
- (21) Cheng, A.; Eng, E. T.; Alink, L.; Rice, W. J.; Jordan, K. D.; Kim, L. Y.; Potter, C. S.; Carragher, B. High Resolution Single Particle Cryo-Electron Microscopy Using Beam-Image Shift. *Journal of Structural Biology* **2018**, *204* (2), 270–275. <https://doi.org/10.1016/j.jsb.2018.07.015>.
- (22) Lindner, J. M.; Cornacchione, V.; Sathé, A.; Be, C.; Srinivas, H.; Riquet, E.; Leber, X.-C.; Hein, A.; Wrobel, M. B.; Scharenberg, M.; et al. Human Memory B Cells Harbor Diverse Cross-Neutralizing Antibodies against BK and JC Polyomaviruses. *Immunity* **2019**, *50* (3), 668-676.e5. <https://doi.org/10.1016/j.immuni.2019.02.003>.
- (23) Matsumoto, S.; Cavadini, S.; Bunker, R. D.; Grand, R. S.; Potenza, A.; Rabl, J.; Yamamoto, J.; Schenk, A. D.; Schübeler, D.; Iwai, S.; et al. DNA Damage Detection in Nucleosomes Involves DNA Register Shifting. *Nature* **2019**, 1. <https://doi.org/10.1038/s41586-019-1259-3>.
- (24) Rabl, J.; Bunker, R. D.; Schenk, A. D.; Cavadini, S.; Gill, M. E.; Abdulrahman, W.; Andrés-Pons, A.; Luijsterburg, M. S.; Ibrahim, A. F. M.; Branigan, E.; et al. Structural Basis of BRCC36 Function in DNA Repair and Immune Regulation. *Mol. Cell* **2019**. <https://doi.org/10.1016/j.molcel.2019.06.002>.



```
#!/bin/bash -noprofile
##### (A) read in metadata #####
. data_connector.sh
##### define additional parameters #####
gain_ref=${stack_source_path}/${name}*gain-ref*
source_stack=${stack_source_path}/${name::${#name}-4}*mrc
copyraw_movie_dir=${destination_path}/movies_raw/
copyraw_micrographs_dir=${destination_path}/micrographs_raw/
copyraw_xml_dir=${destination_path}/xml/
##### create destination folders #####
mkdir -p $copyraw_movie_dir
mkdir -p $copyraw_micrographs_dir
mkdir -p $copyraw_xml_dir
##### define output files #####
copyraw_raw_stack=${copyraw_movie_dir}/${short_name}.mrcs
copyraw_xml=${copyraw_xml_dir}/${short_name}.xml
##### (B) define raw data #####
RAW_FILES copyraw_raw_stack copyraw_xml
if compgen -G "$gain_ref" ; then
    copyraw_gain_ref=${copyraw_movie_dir}/${short_name}_gainref.mrc
    RAW_FILES copyraw_gain_ref
fi
copyraw_log=${copyraw_micrographs_dir}/${short_name}_copyraw.log
copyraw_raw_average=${copyraw_micrographs_dir}/${short_name}.mrc
copyraw_raw_average_fft_png=${copyraw_micrographs_dir}/${short_name}_fft.png
##### (C) define output files #####
FILES copyraw_raw_average copyraw_raw_average_fft_png copyraw_log
##### run processing if files are missing #####
if FILES_MISSING; then
    if compgen -G "$gain_ref" ; then
        echo cp $gain_ref $copyraw_gain_ref
        cp $gain_ref $copyraw_gain_ref
    fi
    cp $source_stack $copyraw_raw_stack
    cp $xml_file $copyraw_xml
    cp ${xml_file}/.xml/.mrc $copyraw_raw_average
    fft_params="--process math.realtofft "
    fft_params+="--meanshrink 7 "
    fft_params+="--process mask.sharp:inner_radius=1 "
    fft_params+=" $copyraw_raw_average "
    fft_params+=" $copyraw_raw_average_fft_png "
    RUN e2proc2d.py $fft_params >> $copyraw_log 2>&1
fi
```

Supplementary Figure 1: Raw data copy script. (A) adapter to read in metadata. (B) Function to define raw data output. (C) Function to define output files.

```
#!/bin/bash --noprofile
#### read input, determine defocus, and extract output parameters #####
<...> # skipped for brevity
##### (A) remove bad quality micrographs from export #####
LIMIT_EXPORT gctf_epa_limit $gctf_input_epa_high $gctf_input_epa_low
##### write Relion star #####
rln_jobtype=CtfFind
rln_jobid=6
printf -v jobfolder "job%03d" $rln_jobid
relion_job_micrographs_dir=$destination_path/$rln_jobtype/$jobfolder/$micrographs_dir
mkdir -p $relion_job_micrographs_dir
rln_alias=gctf
rln_nodetype=1
rln_sturname=micrographs_ctf.star
rln_inputstar=Import/job001/micrographs.star
rln_header=(MicrographName CtfImage DefocusU DefocusV DefocusAngle Voltage \
             SphericalAberration AmplitudeContrast Magnification DetectorPixelSize \
             CtfFigureOfMerit CtfMaxResolution)
##### (B) writing data to Relion star file #####
RELION_WRITE $destination_path $rln_jobtype $rln_jobid $rln_alias $rln_nodetype \
             $rln_sturname $rln_inputstar rln_header[@] \
             $micrographs_dir/${short_name}_DW.mrc \
             $rln_jobtype/$jobfolder/$micrographs_dir/${short_name}.ctf:mrc \
             $gctf_defocus_u $gctf_defocus_v $gctf_defocus_angle 300 0.001 0.07 \
             $magnification $dstep $gctf_epa_cc $gctf_epa_limit

#####(C) export result parameters #####
RESULTS gctf_epa_limit gctf_epa_cc gctf_defocus gctf_defocus_u gctf_defocus_v \
        gctf_astigmatism gctf_defocus_angle gctf_phase_shift
```

Supplementary Figure 2: Data export part of CTF determination script. (A) data selection based on given limits. (B) Creation of Relion pipeline and star files and adding data to Relion star file. (C) Definition of result parameters.